Telephonic Interview, Aug. 21, @ ~~1pm~~ EST
4pm

Dear Examiner Todd,

First off, thank you for granting this telephonic interview. In order to expedite
prosecution, here are the remarks I propose for our discussion. The claims are listed here
for your convenience.

Thank you,

Greg

Please find the following remarks as a proposal for our discussion on Aug. 21.

Listing of Claims:

1.   (Previously Presented) A system comprising:

a plurality of servers organized into one or more failover groups and over which
data is partitioned, each server usually processing client requests for data of a respective
type and processing the client requests for data other than the respective type for other of
the plurality of servers within a same failover group when the other of the plurality of
servers within the same failover group are offline; and,

a master server managing notifications from one or more clients and from the
plurality of servers as to whether servers are offline, the master server verifying whether
a server is offline when so notified, and where the server has been verified as offline, so
notifying the plurality of servers other than the server that has been verified as offline.

2.   (Original) The system of claim 1, further comprising a database storing
data responsive to client requests of any respective type and which has been partitioned
over the plurality of servers, each server caching the data stored in the database
responsive to client requests of the respective type.

3.   (Original) The system of claim 2, wherein each server further temporarily
caches the data stored in the database responsive to client requests other than the

respective type when the other of the plurality of servers within the same failover group are offline.

4.   (Original) The system of claim 1, wherein the one or more failover groups consists of one failover group, such that the plurality of servers are within the one failover group.

5.   (Original) The system of claim 1, further comprising one or more clients sending requests to the plurality of servers.

6.   (Previously Presented) A system comprising:

a plurality of servers organized into one or more failover group, each server usually processing client requests of a respective type and processing the client requests other than the respective type for other of the plurality of servers within a same failover group when the other of the plurality of servers within the same failover group are offline; and,

a database storing data responsive to client requests of any respective type and which is partitioned for caching over the plurality of servers, each server caching the data stored in the database responsive to client requests of the respective type, each server also temporarily caching the data stored in the database responsive to client requests other than the respective type when the other of the plurality of servers within the same failover group are offline.

7.   (Original) The system of claim 6, further comprising a master server managing notifications from one or more clients and from the plurality of servers as to whether servers are offline, the master server verifying whether a server is offline when so notified, and where the server has been verified as offline, so notifying the plurality of servers other than the server that has been verified as offline.

8.   (Original) The system of claim 6, wherein the one or more failover groups consists of one failover group, such that the plurality of servers are within the one failover group.

9.   (Original) The system of claim 6, further comprising one or more clients sending requests to the plurality of servers.

10.   (Previously Presented) A computer-readable medium having instructions stored thereon for execution by a processor to perform a method comprising:

determining whether a data server of a plurality of data servers is in a failover mode;

in response to determining that the data server is not in the failover mode, sending a request by a client to the data server;

determining whether sending the request was successful;

in response to determining that sending the request was unsuccessful,

entering the failover mode for the data server;

notifying a master server that sending the request to one of a plurality of data servers, was unsuccessful;

determining, by a client, a failover server in a failover group, wherein the failover server group is selected from a plurality of servers; and

sending the request to the failover server.

11.   (Original) The medium of claim 10, the method initially comprising determining the data server as one of a plurality of data servers to which to send the request.

12.   (Original) The medium of claim 10, the method initially comprising in response to determining that sending the request was unsuccessful, repeating sending the request to the data server for a predetermined number of times, and entering the failover mode for the data server if sending the request for the predetermined number of times was still unsuccessful.

13.   (Original) The medium of claim 10, the method further comprising in response to determining that the data server is in the failover mode, determining whether the data server has been in the failover mode for longer than a predetermined length of time; and,

in response to determining that the data server has not been in the failover mode

for longer than the predetermined length of time, sending the request to the failover server.

14.   (Original) The medium of claim 13, the method further comprising in response to determining that the data server has been in the failover mode for longer than the predetermined length of time, sending the request to the one of the plurality of data servers;

determining whether sending the request was successful;

in response to determining that sending the request was unsuccessful, sending the request to the failover server;

in response to determining that sending the request was successful, exiting the failover mode for the data server; and,

notifying the master server that sending the request to the data server was successful.

15.   (Currently Amended) A method for performance by a server configured in a failover group, wherein the failover group is selected from a plurality of servers, comprising:

receiving a request from a client by the server;

determining whether the request is of a type usually processed by the server;

in response to determining that the request is of the type usually processed by the server, processing the request by the server;

in response to determining that the request is not of the type usually processed by the server,   determining by the server whether a second server that usually processes the type of the request is indicated as offline;

in response to determining that the second server that usually processes the type of the request is indicated as offline, processing the request by the server;

in response to determining that the second server that usually processes the type of the request is not indicated as offline, sending the request to the second server;

in response to determining that sending the request was unsuccessful, processing the request by the server; and,

notifying by the server a master server that the second server is offline.

16.   (Original) The method of claim 15, further comprising receiving indication from a master server that the second server is online.

17.   (Original) The method of claim 15, further comprising receiving indication from a master server that the second server is offline.

18.   - (Original) A computer-readable medium having instructions stored thereon for performing the method of claim 15.

19.   (Currently Amended) A machine-readable medium having instructions stored thereon for execution by a processor of a master server to perform a method comprising:
    receiving a notification from a client that a server may be offline;
    contacting the server;
    determining whether contacting the server was successful;
    in response to determining that contacting the server was unsuccessful, marking the server as offline; and,
    notifying a failover group of servers selected by the client from a plurality of servers, wherein the client is in failover mode, and wherein the failover group is capable of processing requests for partitioned data of a respective type and partitioned data other than its respective type, other than the server marked as offline that the server is offline.

20.   (Original) The medium of claim 19, the method further comprising periodically checking the server that has been marked as offline to determine whether the server is back online.

21.   (Original) The medium of claim 20, wherein periodically checking the server that has been marked as offline comprising:
    contacting the server;
    determining whether contacting the server was successful;
    in response to determining that contacting the server was successful, marking the server as online; and,
    notifying the plurality of servers other than the server marked as online that the

server is online.

PROPOSED AGENDA FOR DISCUSSION

Rejection of Claims 1-21 under 35 U.S.C. § 112, ¶ 2

Regarding claims 10-14 and 19-21, the Examiner has stated that these claims allegedly omit certain steps, namely, the Examiner has stated: "It is not clear how the client determines the failover server in the failover group" (Office Action, p. 2). The Specification, in the numbered paragraph, provides the following example:

> [0012] A client preferably operates in failover mode as to an offline server for a predetermined length of time. During the failover mode, the client sends requests for data usually handled by the offline server to the failover server that it selected for the offline server. Once the predetermined length of time has expired, the client sends its next request for data of the type usually handled by the offline server to this server, to determine if it is back online. If the server is back online, then the failover mode is exited as to this server. If the server is still offline, the client stays in the failover mode for this server for at least another predetermined length of time.

(emphasis added). Thus, to answer the Examiner's question, the client enters a failover mode, and, during the failover mode the client sends requests for data to the failover server that **it selected for the offline server.** Careful attention must be paid to the verb tense: "client **sends** [present tense] request for data ... to the failover server that it **selected** [past tense] for the offline server." In short, the client determines the failover server in the failover group because it had selected the server as a failover server (hence the use of the past tense verb "selected").

In light of these remarks, claim 10 does not omit any steps, since it explicitly recites "a failover mode." Claim 19 has been amended to add the limitation: "wherein the client is in failover mode."

Regarding claim 15, it was the Applicants' intent to emphasize the point that this "method for performance [was done] by a server configured in a failover group..." (preamble) (emphasis added), however, seeing how this attempted clarification resulted in confusion, the Applicants withdraw this last amendment

and restore the claims to their previous state. The preamble is clear that the method is for performance by a server.

Thus, in light of the above remarks, withdrawal of the rejections for claims 10-14, 19-21, and 15 is respectfully requested.

Rejection of Claims 1-21 under 35 U.S.C. § 102(e)

Claims 1, 6, 10, 15, and 19 are the independent claims. The Applicants address each of these claims sequentially.

Regarding claim 1, the Applicants have submitted that Le et al. does not teach or suggest managing notifications from *both* the clients making requests on the network and from the plurality of servers as to whether the servers are offline (previous Response, p. 7). In the present Office Action, the Applicants have considered the Examiner's remarks in light of Figs. 3 and 4.

The Applicants thank the Examiner for constructing the exemplary scenario in the Office Action, as it was helpful in better understanding the differences between the claimed subject matter and Le et al., but upon reflection, the Applicants conclude that the claimed subject matter cannot be found in Le et al.: "a master server managing notifications from one or more clients **and** from the plurality of servers as to whether servers are offline" (claim 1)(emphasis added).

Next, turning to claim 6, the Examiner observes that "data is inherently partitioned (as Applicant notes in the background of the application, see paragraph 5)" (Office Action, p. 13, ll. 10-11). The notion of inherency and the notion of admitted prior art are two separate issues. First, inherency requires that the inherent elements be *necessarily* taught. *See* MPEP 2112. Applicants submit that data partitioning across servers is not a *necessary* state of events (unless the Examiner's point is that *all data*, at *all times* in the computing world *must be* portioned across servers). The Applicants respectfully disagree with this observation.

Second, the Examiner accurately points out that paragraph 5 discloses "[f]or constantly changing data, the data is more typically partitioned across a number of different web servers." However, this is not all that claim 6 recites: "storing data responsive to client requests of any respective type and which is *partitioned for caching over the plurality of servers*." Thus, claim 6 does not merely recite partitioning of data, but rather "partitioning *for caching....*" When this limitation is read in the context of the entire claim, it becomes evident why this is done: "each server also temporarily caching the data stored in the database responsive to client requests other than the respective type when the other of the plurality of servers within the same failover group are offline." Temporary caching is thus enabled by this limitation.

The Applicants point the Examiner to remarks made in the previous response as to why claim 6 patentably defines over Le et al. (page 8, ll. 4-19).

Thus, in light of the remarks made above and in the previous response, the Applicants respectfully ask for the withdrawal of the rejection of claim 6.

Next, turning to claim 10, the Examiner has noted that "it is not clear how the client can randomly determine another server with which to communicate without a master server or role manager to assist the client in determining the failover server" (Office Action, p. 13, l. 20, p. 14, l. 2). The Applicants point the examiner to the remarks made above addressing the § 112, ¶ 2 rejections. In light of these remarks, reconsideration and withdrawal of the rejection of claim 15 is respectfully requested.

Next, turning to claim 15, the Applicants reiterate the fact that claim 15 recites a method for performance by a server (as opposed to multiple interacting components). The Office Action relies on more than the actions of a server to allegedly anticipate claim 15. For instance, reference is made to steps fulfilled by a "kernel acting with the heartbeat manager." Thus, Le et al. cannot be said to teach or suggest the claimed subject matter. Reconsideration and withdrawal of the rejection of claim 15 is respectfully requested.

Finally, turning to claim 19, Le et al. fails to teach "notifying a failover group of servers selected *by the client*." Again, the above remarks regarding §

112, ¶ 2 explain how this is done. Reconsideration and withdrawal of the rejection of claim 15 is respectfully requested.

Inasmuch claims 2-5, 7-9, 11-14, 16-18, and 20-21 depend either directly or indirectly from independent claims 1, 6, 10, 15, and 19, respectively, they are believed allowable for the same reasons. Withdrawal of the rejection under § 102(e) is therefore earnestly solicited.